HBD clusterizer with built in background subtraction (part 2)

Benjamin Bannier, Ermias Atomssa

HBD Meeting

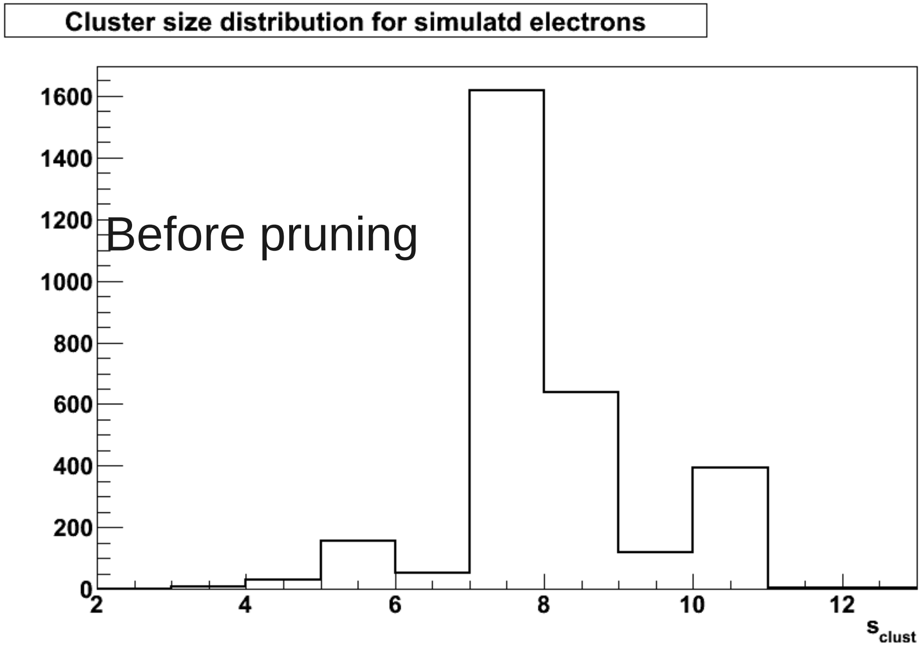
2010.06.30

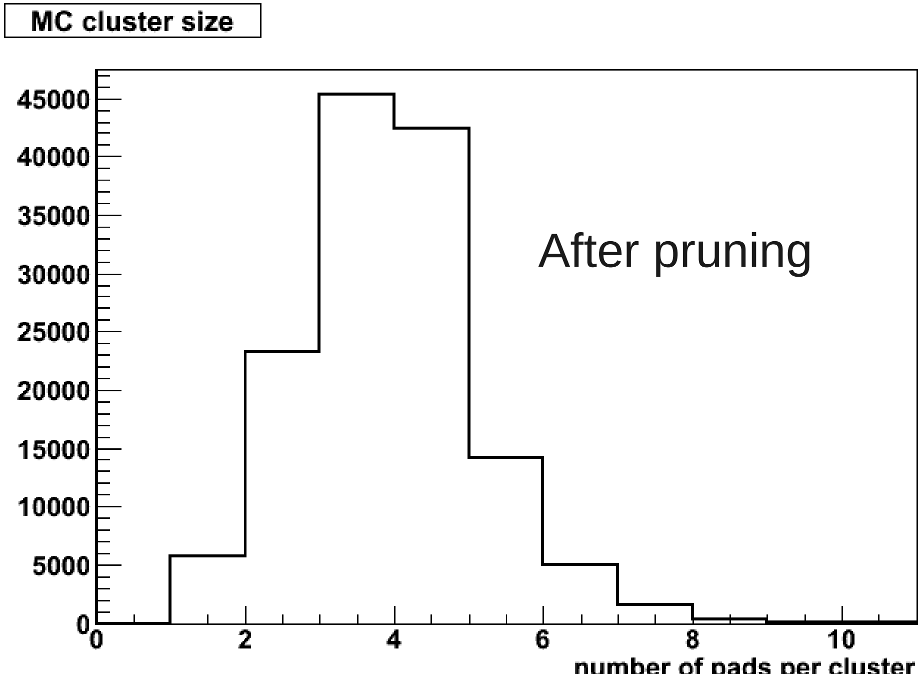
Intro

- Reminder:
    - A clusterization algorithm for the HBD with internal background subtraction based on neighboring pads
- What's new:
    - Cluster size issue in MC solved – pruning
    - Embedding is up and running
    - Efficiency and rejection study vs. centrality

Cluster size issue

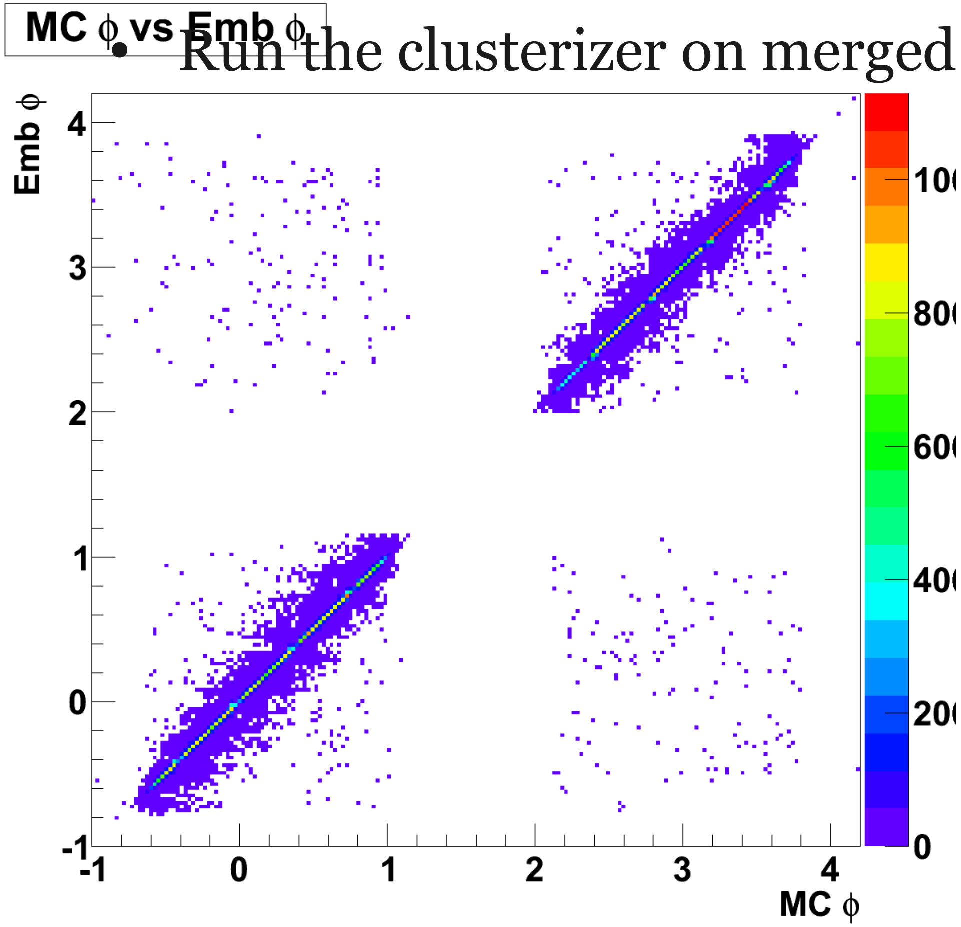
- Because of the way the merging was handled, pure (no background) MC clusters were larger than they should be
- In embedded clusters, this seems to be less of an issue, because peripheral preclusters were rejected at selection (generally higher background meant that the net signal in such clusters was low)
- An additional step of pruning was added to retest **peripheral** pads of a cluster for net signal (background estimated from neighboring pads which are not members of the cluster).
- Peripheral pads are removed if the recalculated net signal is less than zero

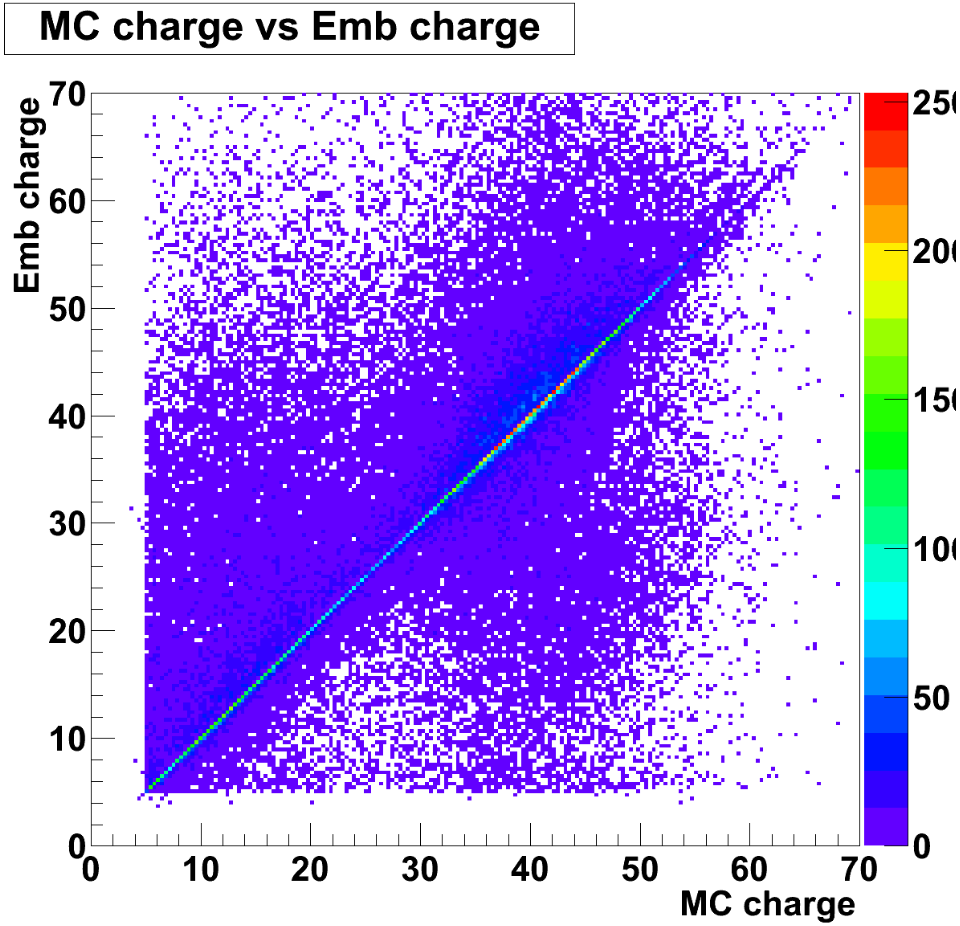




Embedding results

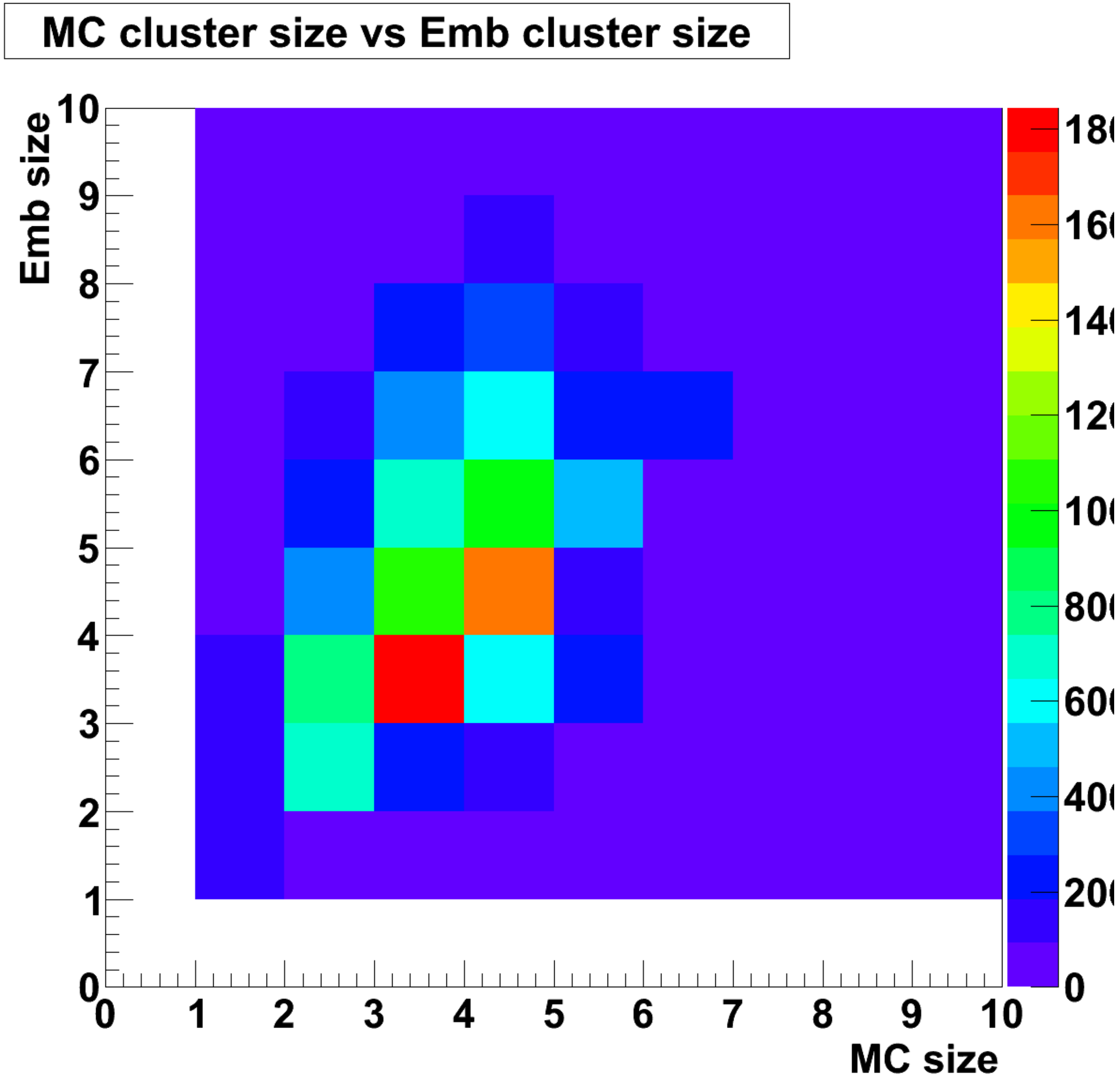
- Embedding can be used to see the effectiveness of a clusterization algorithm with real background
  - Simulate single (or double) electrons
  - Add pad by pad the signal from RD events to the Cerenkov response from MC
  - Run the clusterizer on merged p


MC ϕ vs Emb ϕ

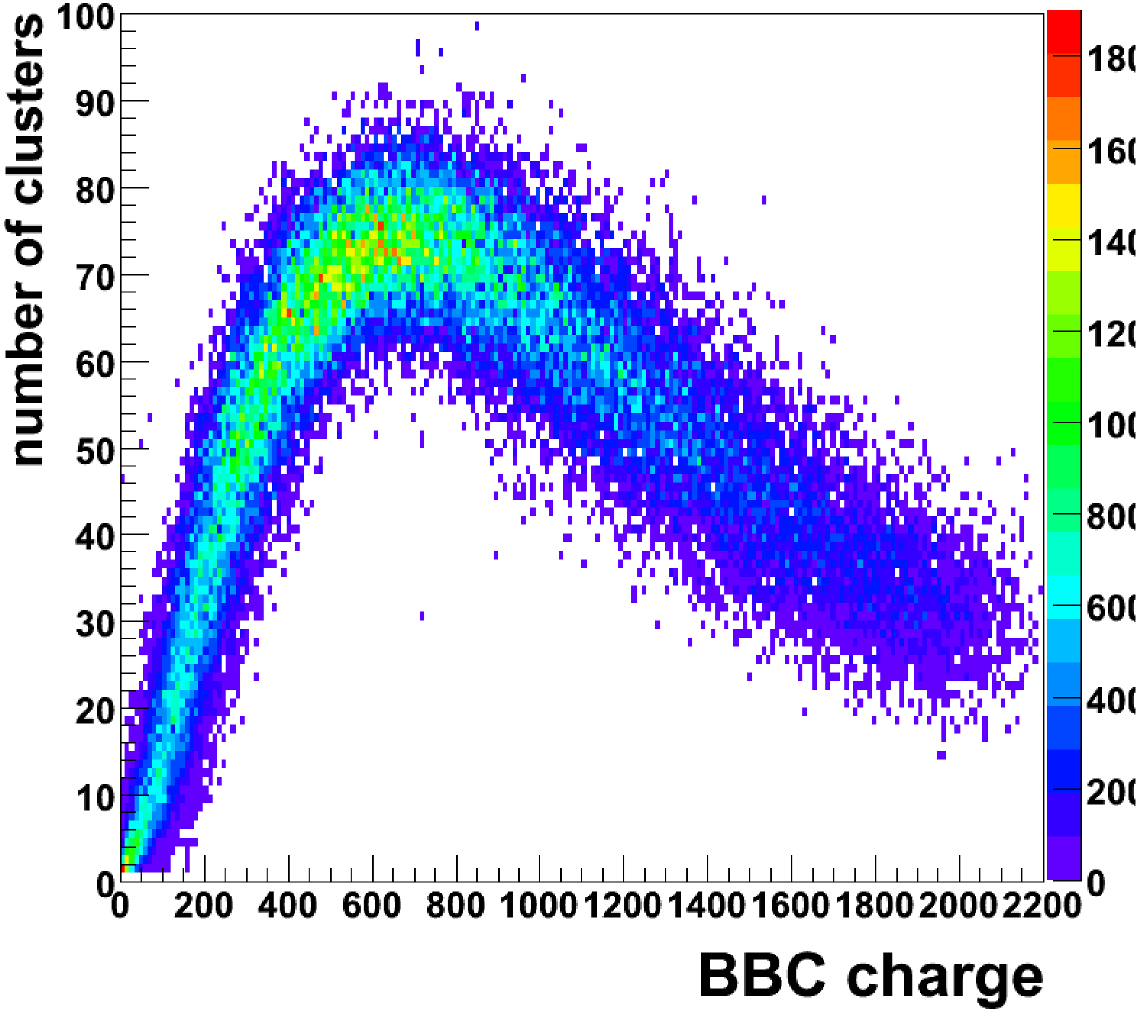

MC charge vs Emb charge

Embedded cluster size

The embedded cluster size correlates well with the MC cluster size



Number of clusters vs. centrality in Embedding

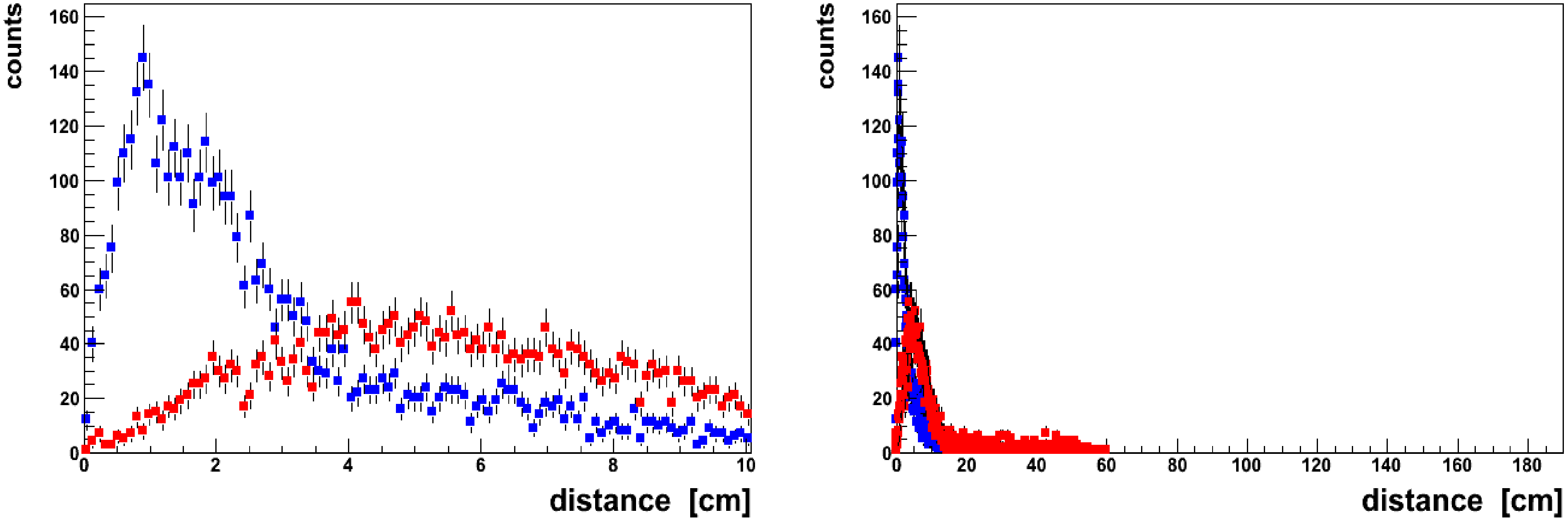


Most events contain no electron => Majority here is “fake” clusters

Saturation & dip for most central events, likely caused by the fact that when most pads are fired, the fluctuations that cause fake clusters start to get buried in scintillation

Distance to the nearest cluster

Centrality = 80-90%

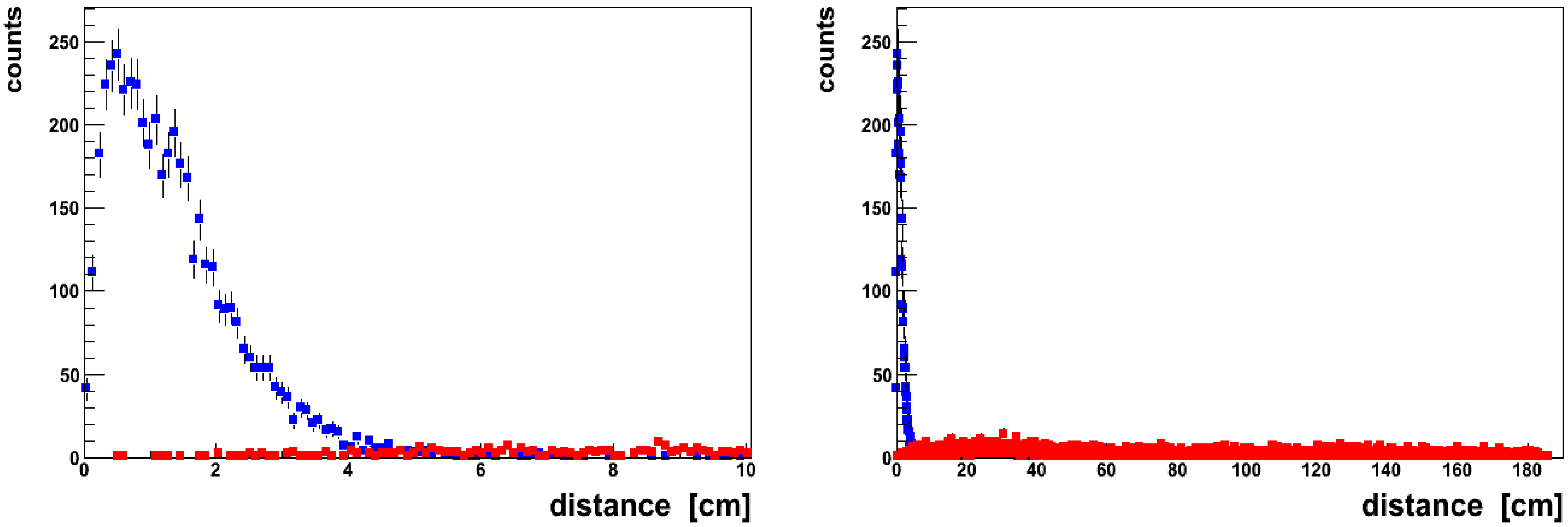


Distance to the nearest cluster from

- The MC electron track projection on HBD surface
- Random point on the HBD surface

Distance to the nearest cluster

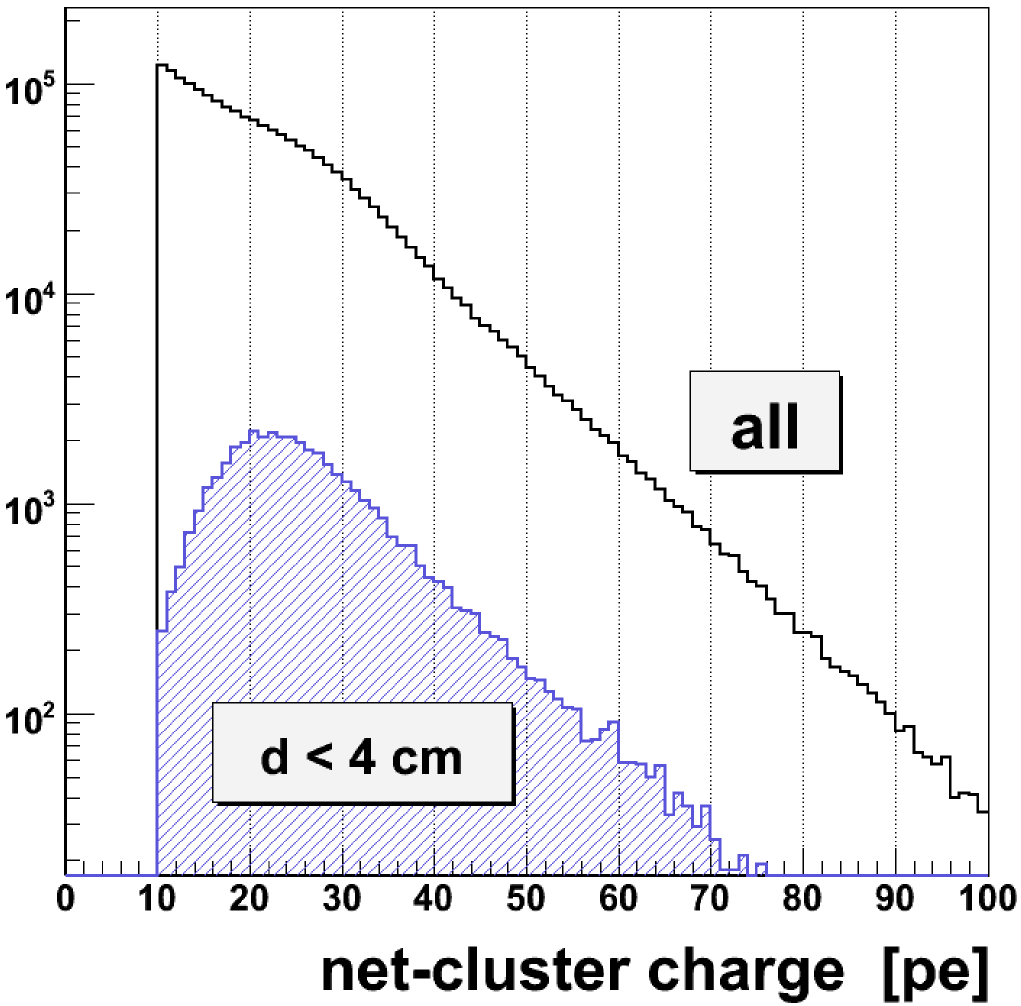
Centrality = 0-10%



Distance to the nearest cluster from

- The MC electron track projection on HBD surface
- Random point on the HBD surface

Cluster net charge distributions



Efficiency and rejection

Rejection at a given cut value d_{cut} can be calculated as

$$R=\frac{\int_{d_{cut}}^{\infty} dN_{rand}}{\int_{0}^{\infty} dN_{rand}}$$

$$dN_{rand} = distance\ of\ nearest\ cluster\ from\ random\ point$$

Efficiency at a given cut value d_{cut} can be calculated as

$$R=\frac{\int_{0}^{d_{cut}} dN_{track}}{N_{trk}}$$

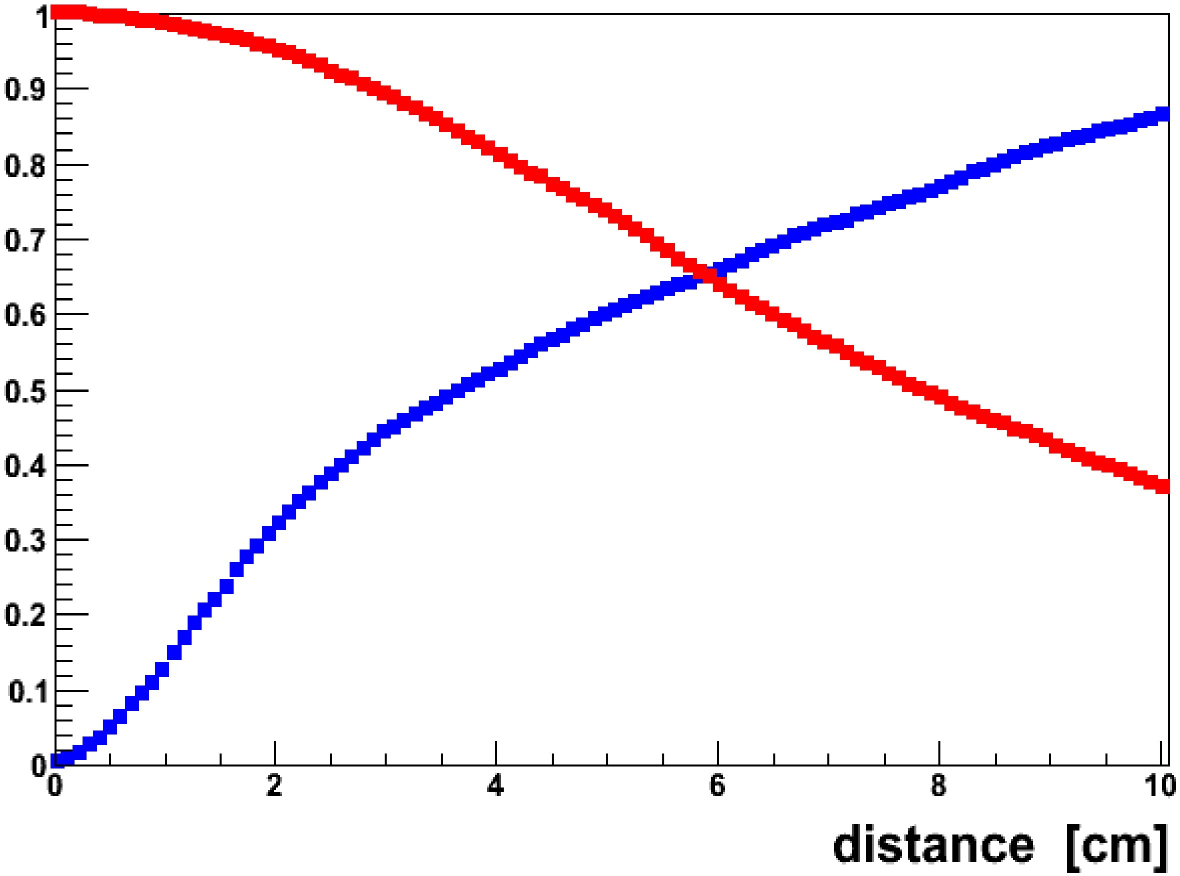
$$dN_{trk} = distance\ of\ nearest\ cluster\ from\ MC\ track\ projection$$

$$N_{trk} = number\ of\ MC\ tracks\ that\ produce\ Cerenkov\ cluster$$

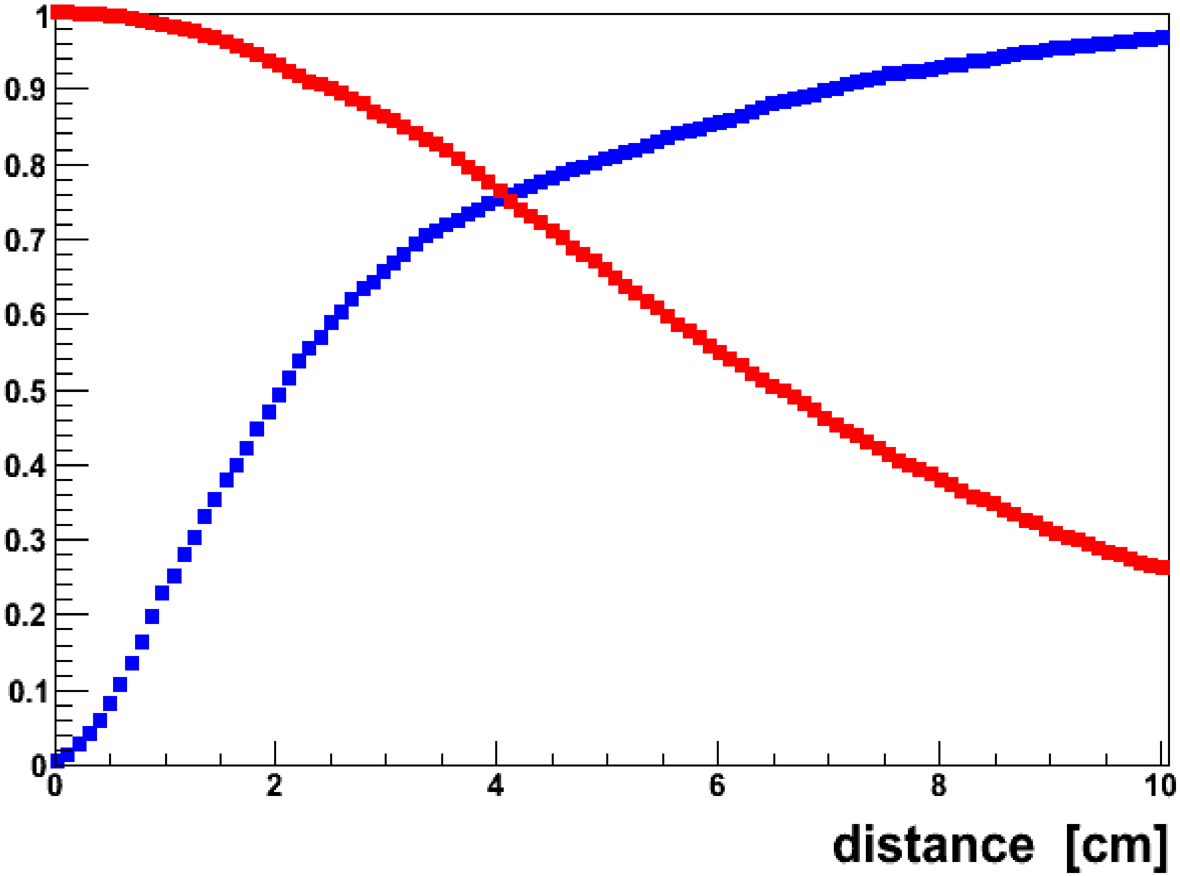
Next slides: efficiency and rejection for 10% centrality bins

- Rad. length of backplane ~ 4x material in Run4
  - if we want to reduce the “late” conversions to 10% of what was already there in Run4 we need to reduce them by a factor of 40, rejection ~97.5%
- Clusterization parameters have not been optimized yet, this is next

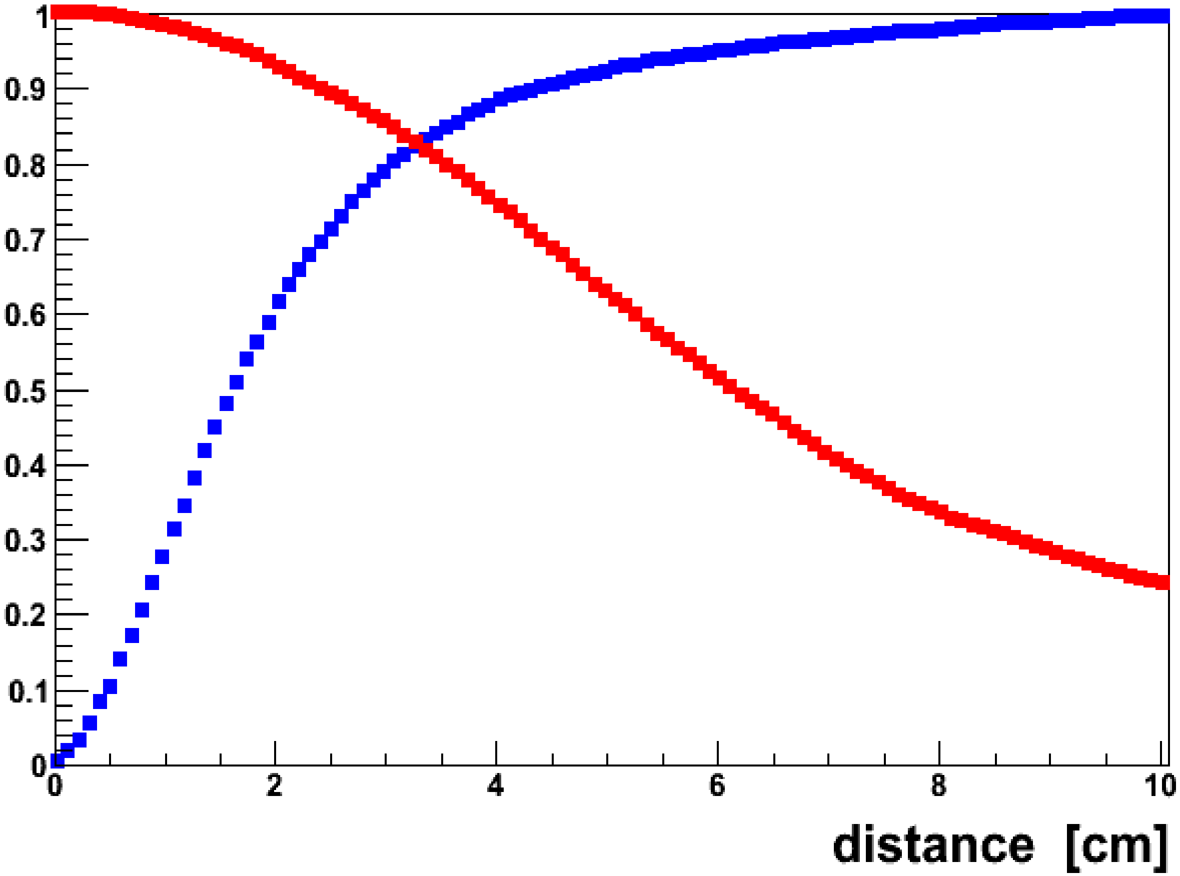
0-10%



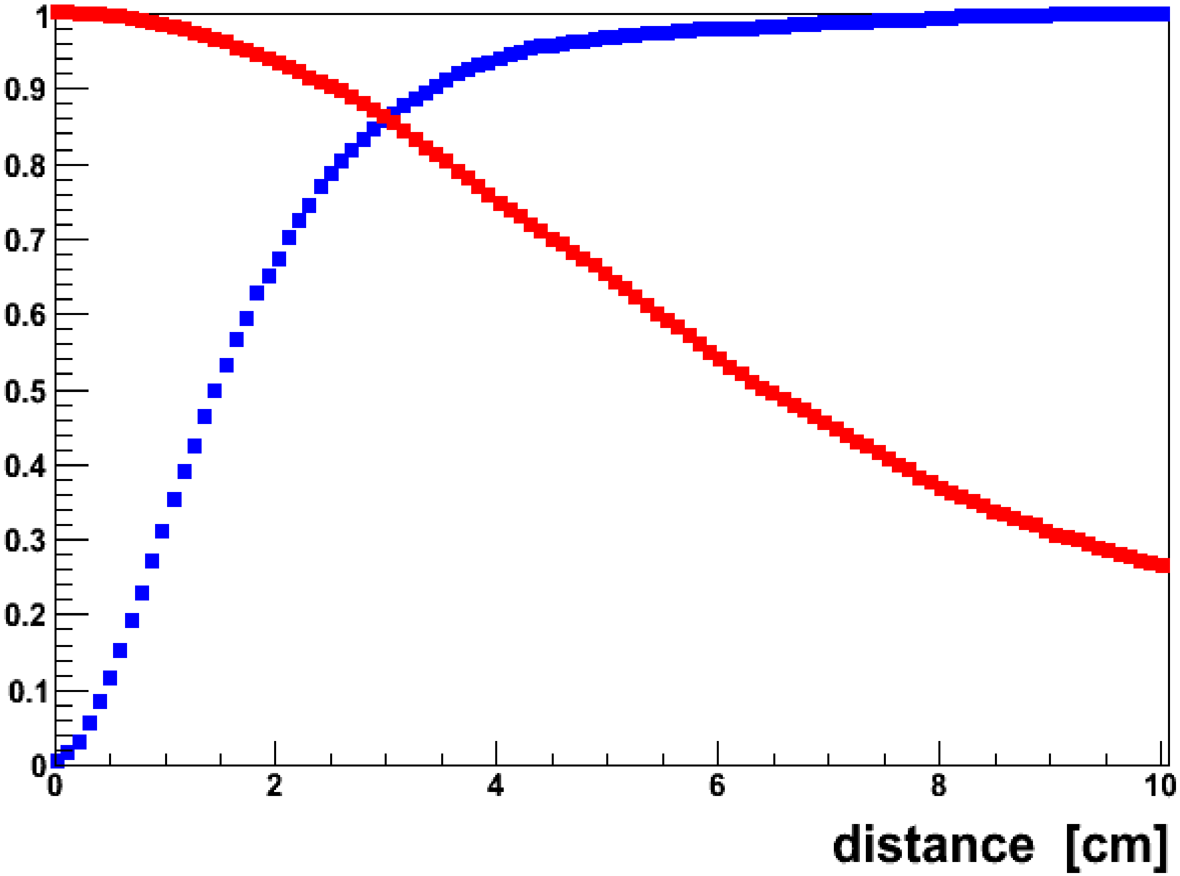
10-20%



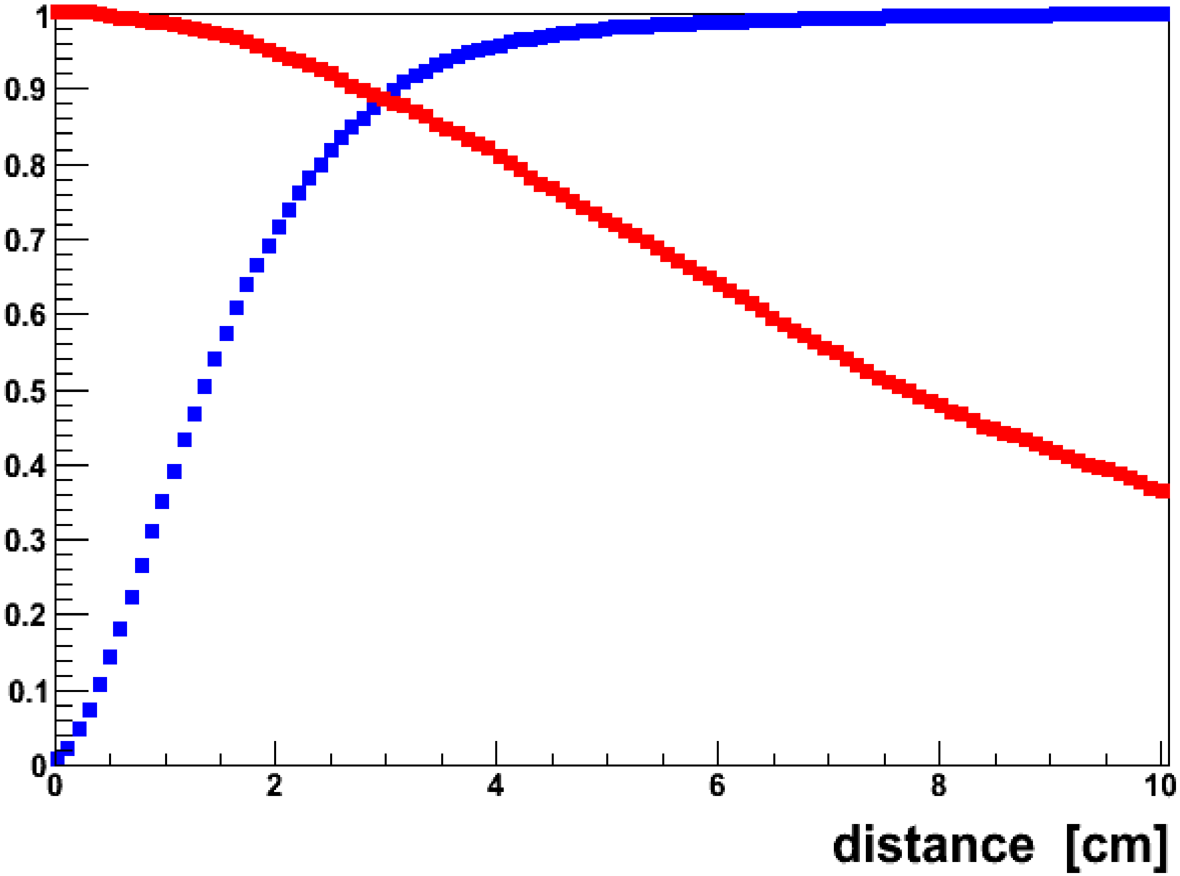
20-30%



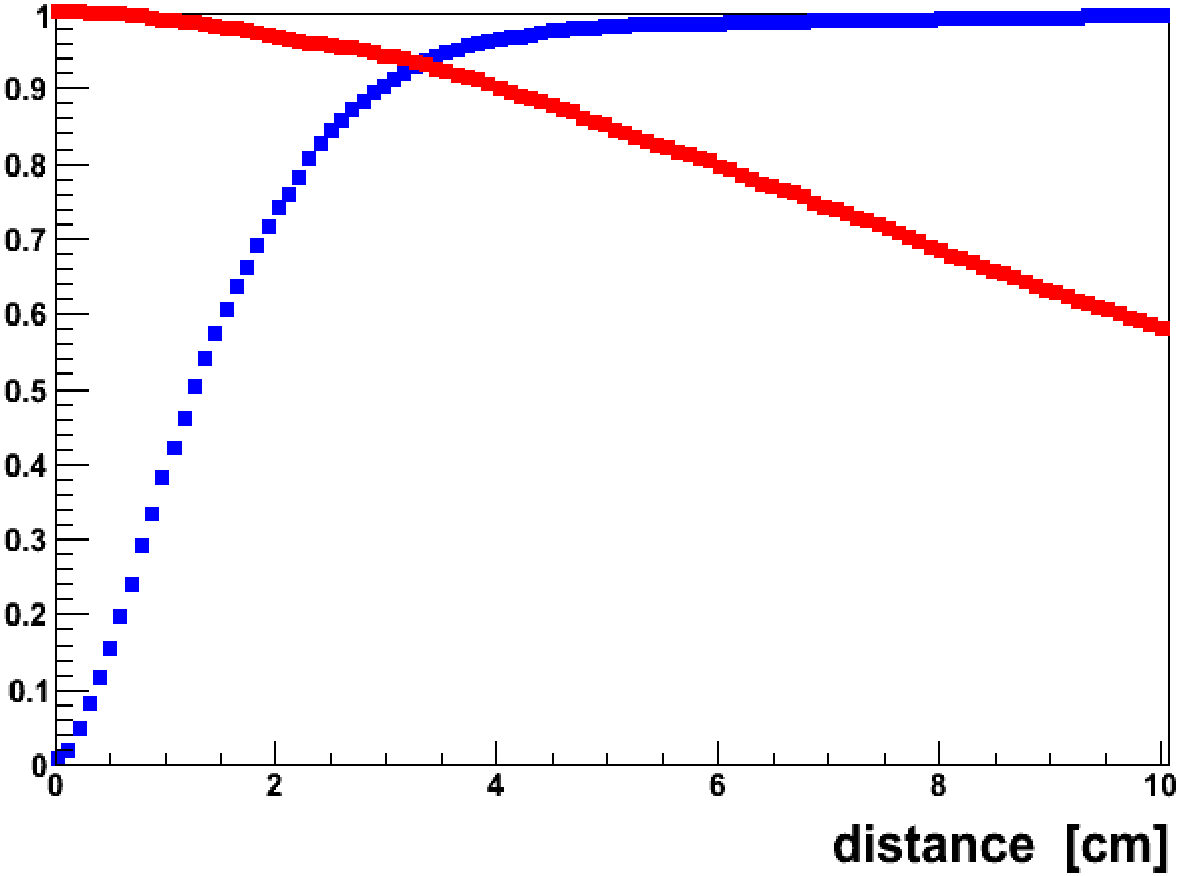
30-40%



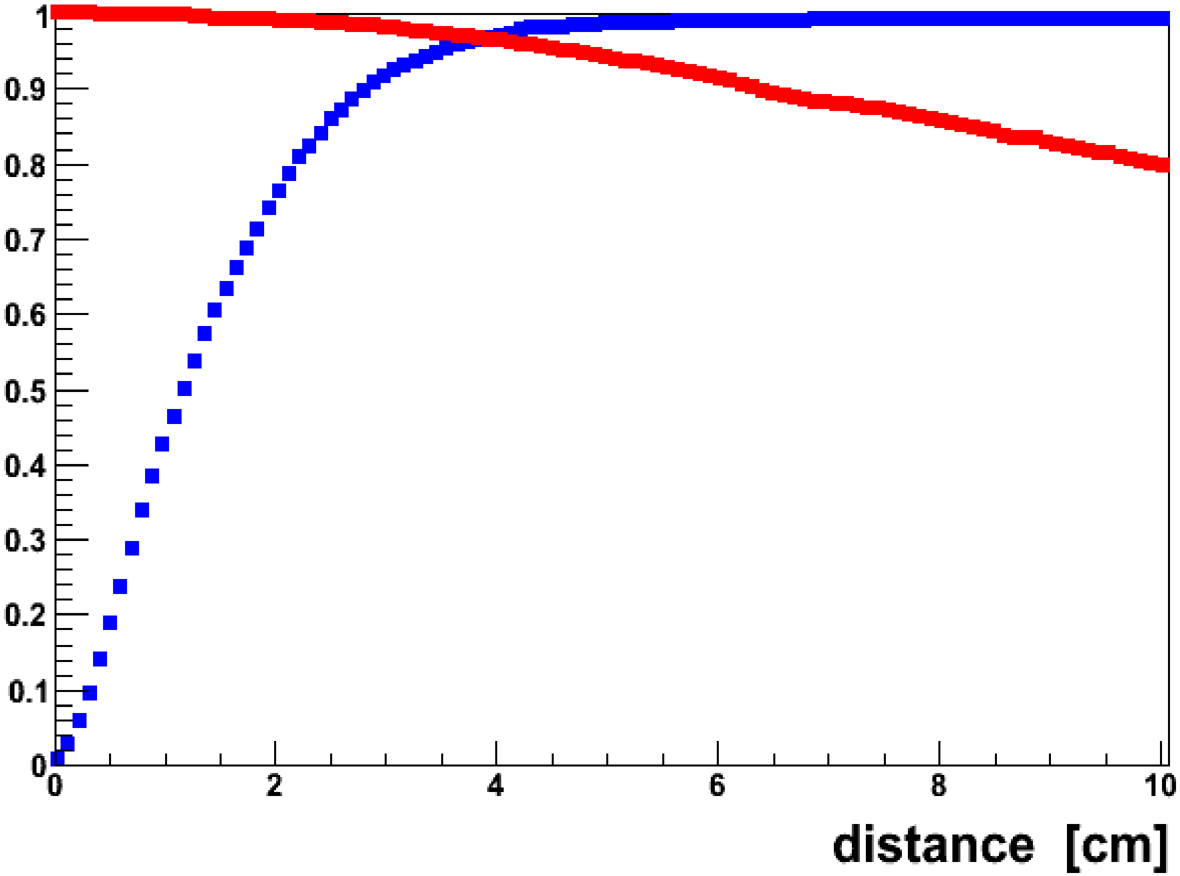
40-50%



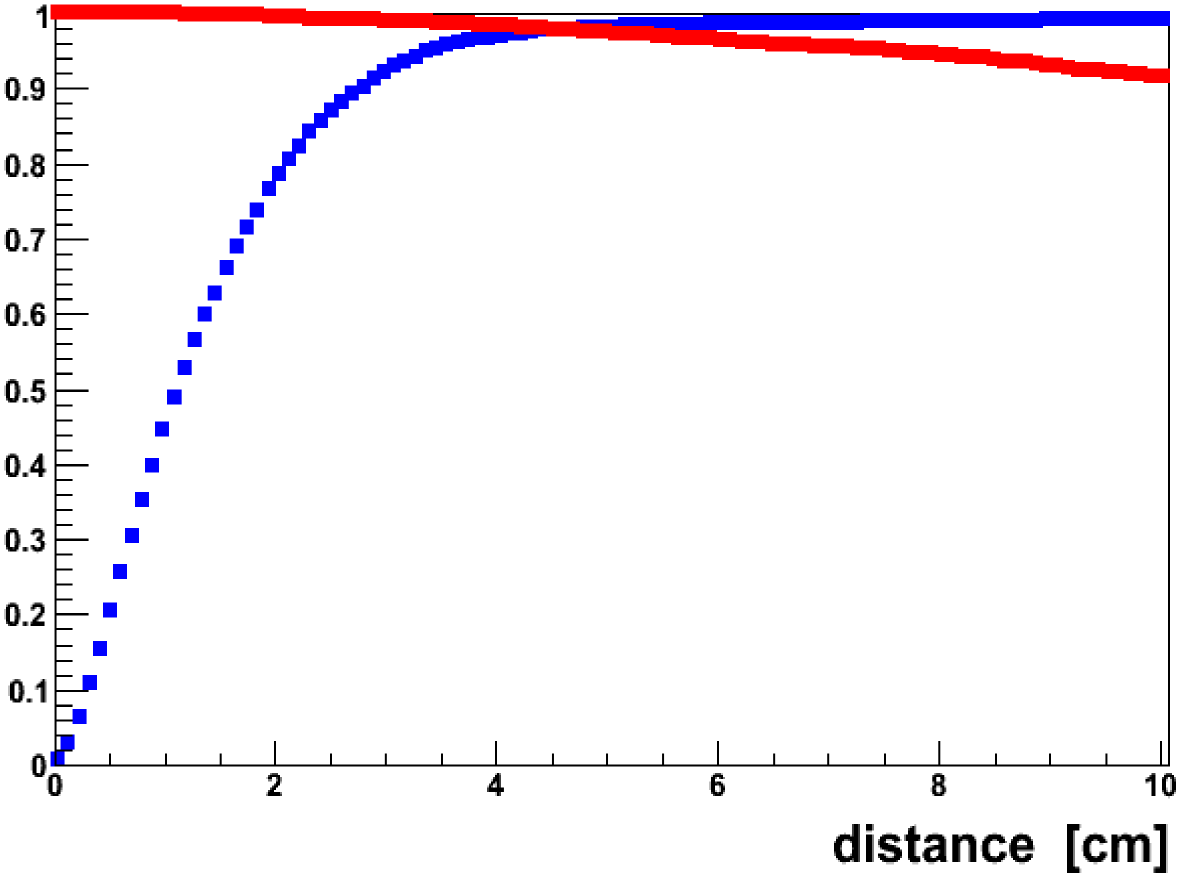
50-60%



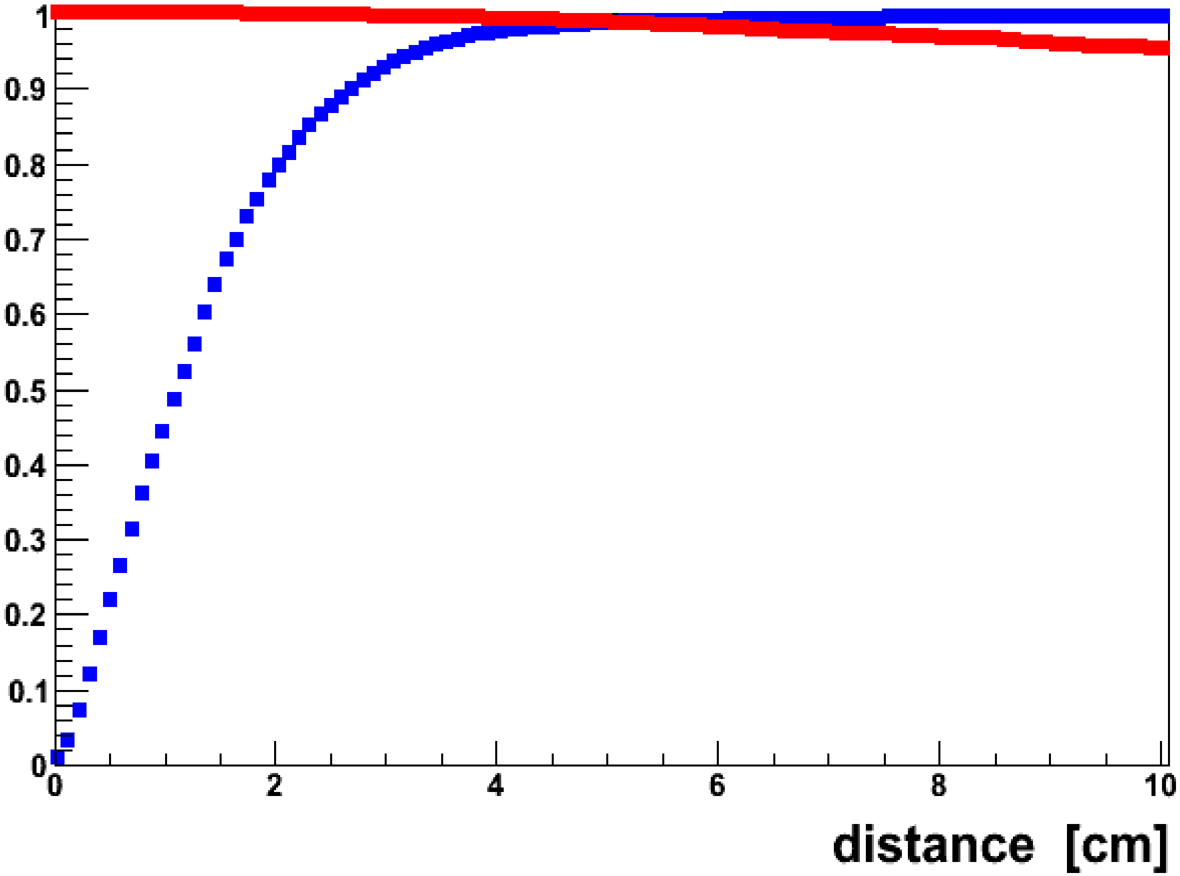
60-70%



70-80%



80-90%

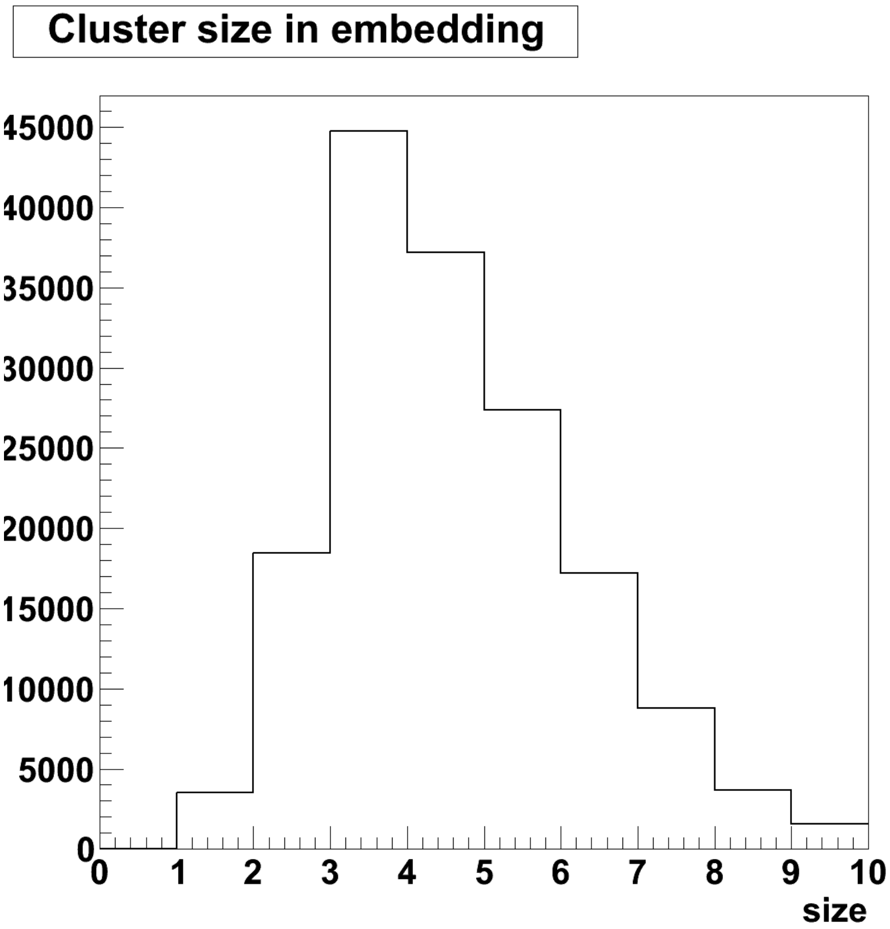


Outro

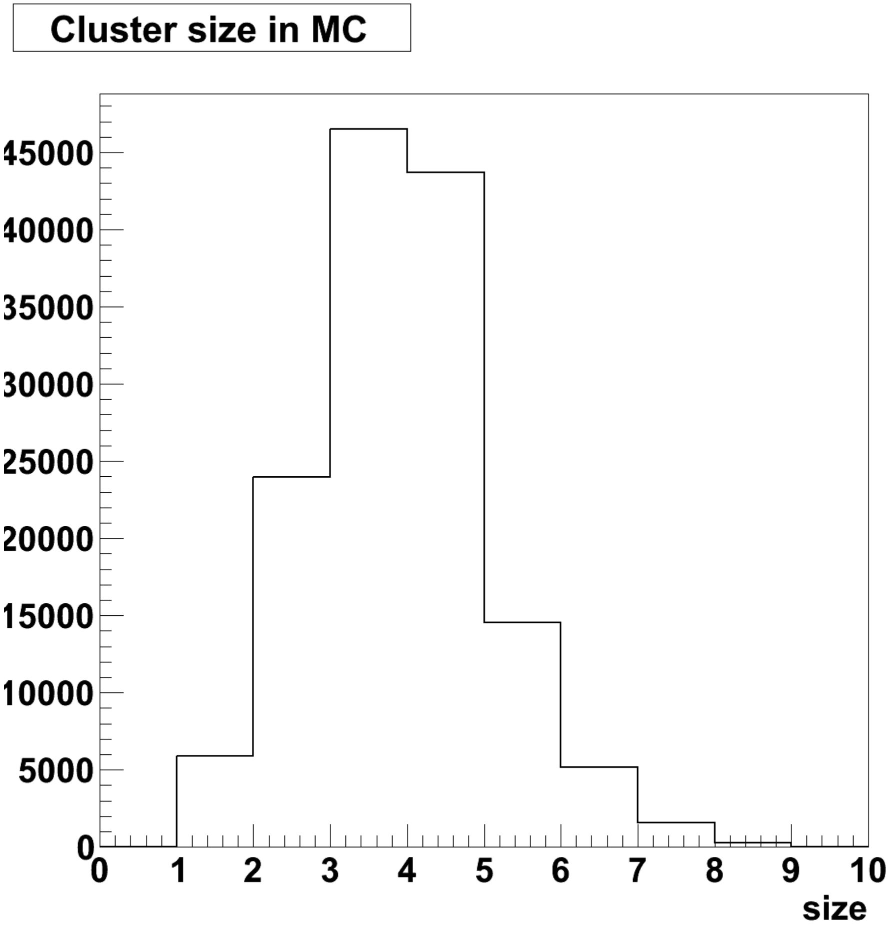
- Cluster size issue fixed
- Embedding works and gives reasonable answers
- Efficiency and rejection
  - Satisfactory for centrality > 30%
  - Less so for more central events
    - Further improvement possible by tuning clusterization cutoffs
- Ongoing
  - Double hit response
  - Hadron rejection
  - Fine tune the clusterizer for different centralities

Backup

Cluster Sizes in MC and embedded MC



Cluster size in embedding



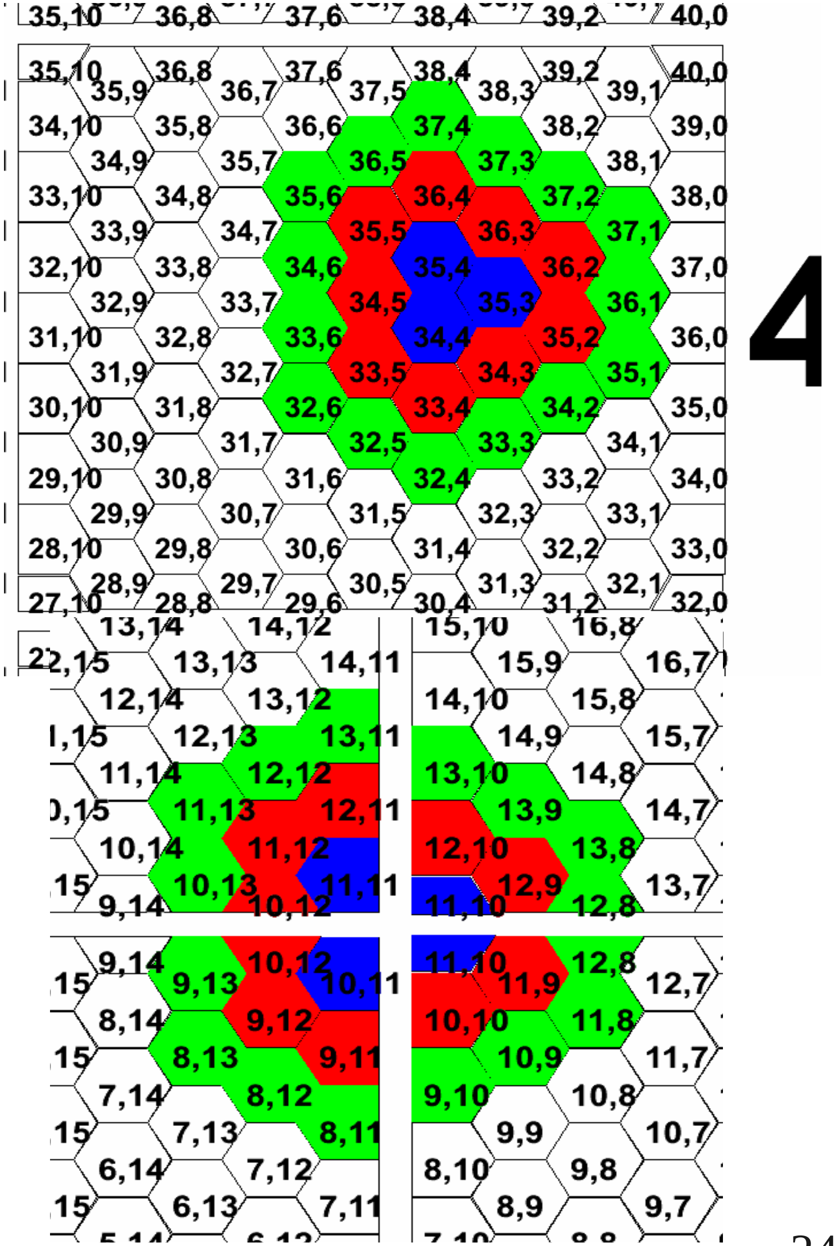
Cluster size in MC

Intro 2: A new clusterization algorithm

- Better of the two worlds:
  - Like Weizmann clusterizer: two steps, preclusterization and merging.
    - But, before merging there is a control step where preclusters are selected based on a few criteria
  - Like HnS clusterizer: preclusters are triplets, most natural shape for the hexagonal symmetry of the HBD pads
    - It doesn't need to depend on the projection of electrons even in high background environment. Though this information can be used if needed.
- And some more....
  - At the preclusterization step, a local background subtraction is internally (without the use of parametrization) applied.
    - This is done by estimating the background level from neighboring pads of the precluster. There seem to be (cf slide 5) reasonable correlation to warrant this
  - After merging, the final cluster's background is subtracted using neighboring pads
  - For this reason, will refer to the new clusterizer as of LBS (local background subtraction) method

Preclusterization

- First step of the algorithm is the selection of preclusters.
  - Candidates for preclusters are all possible compact triplets in the HBD (def. All members sharing a single edge with every other neighbor)
  - Preclusters have
    - first neighbors
    - and second neighbors.
  - And they cross borders
  - They have the following properties:
    - Charge & area of Members
    - Charge & area of 1st & 2nd neighbors
    - Net signal in the “member” zone
    - “Shape” meaning distribution of net



Merging and post merging

- Overlapping preclusters
    - Share atleast one pad
- Final clusters
    - Lump together pads from all overlapping groups of preclusters
- Local bkg. subtraction
    - Merged clusters have 1st and 2nd neighbors just like preclusters
    - 1st and 2nd neighbor charge is used to estimate background to subtract from the members of merged cluster
- Cluster track association
    - Nothing new here, based on proximity just like in Wis & HnS